



Utveckling av en ny webbplats för Skogsvårdsföreningen södra skogsreviret

Daniel Broman

Examensarbete
Informations- och medieteknik
2018

EXAMENSARBETE	
Arcada	
Utbildningsprogram:	Informations- och medieteknik
Identifikationsnummer:	6619
Författare:	Daniel Broman
Arbetets namn:	Utveckling av en ny webbplats för Skogsvårdsföreningen Södra skogsreviret
Handledare (Arcada):	Andrej Shcherbakov
Uppdragsgivare:	Södra skogsreviret rf
<p>Sammandrag:</p> <p>Examensarbetet behandlar valet av byggstenar i samband med utvecklingen av Södra skogsreviret rf:s webbsida. Webbsidan utvecklades i WordPress som är ett mycket modulbaserat innehållshanteringssystem och därför är det viktigt att man noga väljer ut vilka byggstenar som ska användas. I arbetet tas upp de kravspecifikationer som Södra skogsreviret rf gett och på basis av dem diskuteras vilken byggsten som lämpar sig bäst för en specifik uppgift. Syftet med arbetet är att för läsaren förenkla urvalsprocessen av byggstenar för en ny webbsida med moderna egenskaper samt att ge läsaren en bred vision över vad som krävs av en modern webbsida. Arbetet har begränsats så att det inte diskuterar sökmotoroptimering, standard-procedurer som installation av WordPress och hur alla layout-filerna är uppbyggda. Byggstenarna har valts ut baserat på information från öppna källor så som recensioner och information som finns på varje enskild byggstens hemsida. Genom att kombinera informationen och testande i praktiken har de byggstenar som lämpat sig bäst för uppgiften valts ut. Resultatet presenteras genom att de huvudsakliga kravspecifikationerna tas upp och i samband med implementeringens presentation diskuteras hur en eller flera byggstenar har använts för att uppfylla ett specifikt krav.</p>	
Nyckelord:	WordPress, Responsivitet, Användarvänlighet, Insticksprogram, Södra Skogsreviret rf
Sidantal:	36
Språk:	Svenska
Datum för godkännande:	

DEGREE THESIS	
Arcada	
Degree Programme:	Information and media technology
Identification number:	6619
Author:	Daniel Broman
Title:	Development of a new website for Forest management association Södra skogsreviret
Supervisor (Arcada):	Andrej Shcherbakov
Commissioned by:	Södra skogsreviret rf
<p>Abstract:</p> <p>This thesis covers the choice of modules while developing a new website for Södra skogsreviret rf. The website was developed using WordPress which is a very module-based content management system and that's why it's important to carefully select what modules to use. The thesis brings up the specification of requirements that Södra skogsreviret rf gave to me and based on them it is discussed which module that is best suited for a specific task. The purpose of the thesis is to simplify the choice of modules for the reader to develop a modern website and to give the reader a broader vision what is required to develop a modern website. The thesis is restricted so that search engine optimization, standard procedures such as installing WordPress and how the code looks in the layout files isn't discussed. The modules have been selected based on information from open sources such as reviews and information that is shown on the modules own pages. By combining that information and testing it practically the modules that suits the tasks the most has been chosen. The result is presented by bringing up the main requirement specifications and while discussing the implementation it's presented how one or more modules has been used to fulfill a specific requirement.</p>	
Keywords:	WordPress, Responsivity, User-friendliness, Plugin, Södra skogsreviret rf
Number of pages:	36
Language:	Swedish
Date of acceptance:	

INNEHÅLL

1	Inledning.....	8
1.1	Bakgrund	8
1.2	Södra skogsreviret rf	8
1.3	WordPress.....	9
1.4	Syfte och mål.....	9
1.5	Avgränsningar	10
2	Metod.....	10
2.1	Val av teoretiskt material och arbetssätt	10
2.2	Metod för datainsamling	11
2.3	Metod för utvärdering av resultatet.....	12
3	Webbsidans kravspecifikationer	12
3.1	Navigering	12
3.2	Startsidan	14
3.3	Tvåspråkighet.....	14
3.4	Elektroniska formulär för insamling av information	15
3.5	Responsivitet	15
3.6	Självadministrering av webbsidan	16
4	Val av byggstenar	16
4.1	Val av WordPress tema.....	16
4.2	Val av insticksprogram	17
4.2.1	<i>Advanced Custom Fields Pro</i>	17
4.2.2	<i>Polylang</i>	18
4.2.3	<i>Ninja Forms</i>	18
4.2.4	<i>Slider Revolution</i>	19
4.2.5	<i>Post Types Order</i>	19
4.3	Bootstrap	19
5	Resultat och Implementering av krav.....	20
5.1	Navigering	20
5.2	Startsidan	22
5.3	Tvåspråkighet.....	27
5.4	Elektroniska formulär för insamling av information	29
5.5	Responsivitet.....	30
5.6	Självadministrering av webbsidan	32

5.7	Sammanfattning av resultat.....	33
6	Diskussion och slutsats	34
	Källor	35

Figurer

Figur 1. Skogsrevirets område.....	9
Figur 2. Den planerade strukturen på webbsidan.	13
Figur 3. Footerns struktur.	13
Figur 4. Fältet som man kan lägga till en slider med.	22
Figur 5. Vyn då man besöker startsidan.	23
Figur 6. Startsidans sektion som visar huvudtjänster.	25
Figur 7. Startsidans sektion som visar nyheter.	26
Figur 8. Startsidans kontaktsektion.	27
Figur 9. Polylangs användargränssnitt för språkval på editeringssidan.	28
Figur 10. Utseendet på en undersida före(vänster) och efter (höger) responsiviteten fixats.	31
Figur 11. Innehåll på personalsida efter bildförhållandet justerats.	32

TERMINOLOGI

Innehållshanteringssystem (Content Management System - CMS): Program för innehållshantering på en webbplats. WordPress är ett CMS (IT-ord 2018a)

HTML: Hypertext Markup Language, det språk som används för att skriva webbsidor (IT-ord 2018e).

CSS: Cascading Style Sheets (IT-ord 2018b)

PHP: Hypertext Preprocessor. Skriptspråk som används för att formatera dynamiska webbsidor. (IT-ord 2018f)

Insticksprogram: Program som tillför nya funktioner till ett annat system och som, när det har installerats, fungerar som en del av det andra systemet. (IT-ord 2018c)

Kravspecifikation: Beställning av ett IT-system eller program med beskrivning av begärda funktioner och egenskaper. (IT-ord 2018d)

Shortcode: WordPress kod som tillåter skapa avancerade element med endast en rad (WordPress support 2018).

1 INLEDNING

1.1 Bakgrund

Södra skogsreviret rf (Vidare i texten – Skogsreviret eller Föreningen) var i behov av en ny webbsida då den gamla sidan var en del av den riksomfattande sidan som gäller hela Finlands skogsrevir. I och med detta ville ledningen ha en egen, mer personlig webbsida som dessutom skulle vara mera användarvänlig och responsiv än den gamla webbsidan. Jag träffade Skogsrevirets ledning första gången i Lovisa hösten 2017 där vi gick igenom vad som skulle behöva göras.

Jag ville att webbsidan skulle bli så proffsig som möjligt och motsvara Skogsrevirets förväntningar så mycket som möjligt så i och med min tidigare erfarenhet i användandet av WordPress tyckte jag det var det smartaste valet av innehållshanteringssystem.

1.2 Södra skogsreviret rf

Skogsvårdsföreningen södra skogsreviret erbjuder skogsvårdstjänster åt sina medlemmar i södra Finland. Några exempel på utbudet är virkeshandel, skogsvård, skogsbruksplanering och skogscertifiering. Skogsreviret har totalt fyra stycken kontor som tillsammans styr hela området (Figur 1). Föreningen har idag ca 4400 medlemmar men endast 10% av medlemmarna kontaktar Föreningen via webben. Medelåldern på medlemmar är 55 år så deras tekniska färdigheter varierar. Utvecklingen går mot att medlemmar småningom blir yngre så tack vare generationsväxlingar ökar kontakten via webben. Dagens medlemmar kontaktar Föreningen främst per telefon men kontaktuppgifter söks från webbsidan. Eftersom Skogsreviret även har utländska samarbetspartners är det viktigt att deras webbsida hålls uppdaterad.



Figur 1. Skogsrevirets område.

1.3 WordPress

WordPress är ett innehållshanteringssystem som förlitar sig på PHP och MySQL för att lagra och skriva ut information. WordPress grundades år 2003 av Mike Little och Matt Mullenweg och används idag av ca 30% av webben. Det är byggt så att nybörjare kan börja använda det utan större problem, men är även väldigt kraftigt för erfarna utvecklare. (WordPress codex 2018)

Alternativ som webbsidans projekt skulle kunna bygga på är Joomla! eller Drupal. Jag valde dock att inte använda dem då de är gjorda för att upprätthålla större varianter av webbsidor (Astra 2018). Båda alternativen är även svårare att upprätthålla för nybörjare som Skogsrevirets ledning och underlättar då processen för mig att lära dem hur man lägger till och tar bort innehåll. Jag har även mest erfarenhet av WordPress.

1.4 Syfte och mål

Syftet med detta examensarbete är att förenkla för läsaren urvalsprocessen av byggstenar för en ny webbsida med moderna egenskaper. De byggstenar som kommer att diskuteras är WordPress tema och insticksprogram.

Syftet med arbetet är också att läsaren ska få en bred vision över vad som krävs för att skapa en modern webbsida. Jag kommer att förklara hur jag själv utfört utvecklingen av Skogsrevirets webbsida och vilka byggstenar jag använt i processen.

Jag redovisar för de viktigaste egenskaperna på den nya webbsidan som beställaren vill ha och den primära frågan som mitt arbete svarar på är:

- Vilka färdiga WordPresslösningar kan underlätta framställningen av en webbsida med de beställda egenskaperna?

1.5 Avgränsningar

Jag har bland annat lämnat bort standardprocedurer som installation av WordPress, installation av tema och installation av diverse insticksprogram. Detta kan man vid behov bekanta sig med i respektive manualer.

Några andra saker som jag inte kommer att gå in på är:

- Sökmotoroptimering då det är ett väldigt brett område.
- Hur layout filerna är kodade, jag kommer endast att inkludera några kodexempel.
- Analys av hur säker webbsidan är mot inbrott.

2 METOD

I detta kapitel kommer jag att gå igenom mina metodval och mitt arbetssätt. Jag kommer att förklara även hur jag utvärderat mitt resultat.

2.1 Val av teoretiskt material och arbetssätt

Det teoretiska materialet som jag kommer att använda mig av finns i öppna källor online och i akademiska databaser. Materialet inkluderar WordPress egna sidor, online forum som t.ex. Stack Overflow och bloggar.

Då problem uppstår kommer jag att vända mig till online Q & A-samhällen inklusive Stack Overflow för att hitta inlägg där personer som har haft liknande problem eller samma problem som mig själv. Utgående från informationen jag hittar så kommer jag att pröva olika lösningar tills problemet är löst. Jag studerar noggrant WordPress egen dokumentation och följer dennes rekommendationer.

2.2 Metod för datainsamling

För att samla in kravspecifikationer för den nya sidan organiserade jag möten med Skogsrevirets ledning. Under mötena gjorde jag anteckningar som sedan användes för att sammanställa en lista med kravspecifikationer. Jag kommer att presentera listan i kapitel tre.

Insamling angående val av byggstenar kommer att ske genom att söka alternativ på insticksprogram med liknande egenskaper från WordPress sidor. Sedan jämför jag de insticksprogram som har liknande egenskaper för att bedöma vilket program som lämpar sig bäst för en specifik uppgift.

Informationen som jag samlar in för att jämföra de olika alternativen kommer från insticksprogrammets officiella sidor, diskussionsforum samt blogginlägg som gör en jämförelse mellan två eller flera byggstenar. En lista på källor som jag använt finns i källförteckningen.

För att analysera och tolka rekommendationer som jag samlar in angående val av byggstenar kommer jag först att jämföra informationen från de olika programmets sidor. De egenskaper som jag kommer att rikta in mig på är att kolla betyg, recensioner, datum då insticksprogrammet senast uppdaterats samt vilken version av WordPress som programmet är kompatibelt med. Med hjälp av den informationen filtrerar jag bort de program som inte har speciellt högt betyg eller positiva kommentarer. Om programmet inte uppdaterats på flera månader och inte stöder den nyaste versionen av WordPress kommer jag också att filtrera bort det.

2.3 Metod för utvärdering av resultatet

Jag presenterade några prototypversioner av webbsidan åt Skogsrevirets ledning. Ledningen har även fått testat t.ex. kontaktformulären och inmatning av information för att lättare kunna bedöma webbsidan. De har sedan tillsammans kommit fram till ett beslut om webbsidan motsvarar kraven eller inte. Webbsidan kunde konstateras vara klar då Skogsrevirets gemensamma åsikt var att resultatet motsvarar kraven.

För att kunna bedöma vilka byggstenar som lämpar sig bäst i utvecklingen av webbsidan kommer jag att jämföra den insamlade informationen på de olika insticksprogrammen som jag gjort. Sedan kommer jag att testa programmen i praktiken för att sist och slutligen bedöma vilken som lämpar sig bäst. De program som uppfyller kravspecifikationerna kommer att användas.

3 WEBBSIDANS KRAVSPECIFIKATIONER

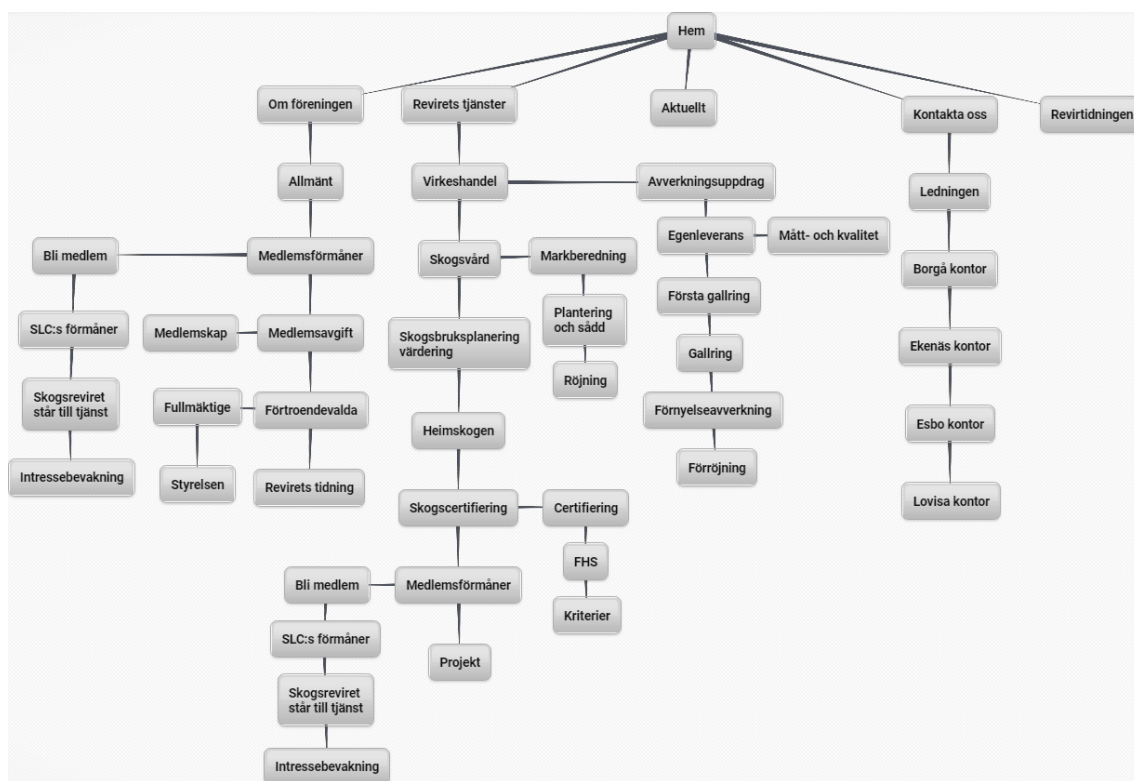
I detta kapitel kommer jag att beskriva de olika kravspecifikationer som Skogsreviret gett åt mig.

3.1 Navigering

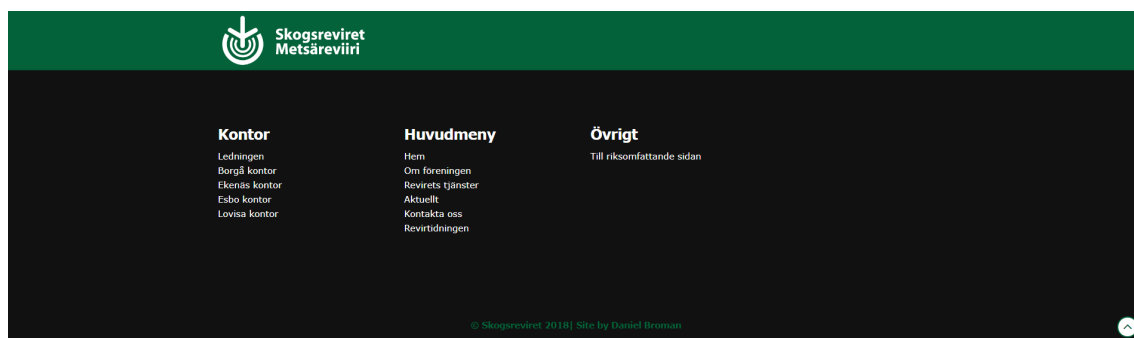
Som nämnt tidigare så är en del av fokusen i detta projekt att göra navigeringen så användarvänlig som möjligt. Eftersom Föreningen hade en stor del information som skall visas på olika sidor så funderade vi gemensamt ut en vettig struktur på föreningens nya webbsida (Figur 2).

Från huvudmenyn skall man kunna nå alla skapade sidor på webbsidan och under en stor del av alla undersidor skall även finnas en sidomeny, till höger om innehållet. Där listas alla sidor som faller under samma huvudämne, t.ex. alla sidor under "Om föreningen" finns i samma sidomeny. Ovanför huvudmenyn ville man att det skulle placeras kontaktuppgifter och en sökruta.

Längst ner på sidan finns footern som enligt kraven skulle ha länkar till alla fyra kontor, huvudmenyns länkar samt en länk till den riksomfattande webbsidan (Figur 3).



Figur 2. Den planerade strukturen på webbsidan.



Figur 3. Footerns struktur.

3.2 Startsidan

Startsidan är oftast den viktigaste delen på en webbsida då meningen med den är att rikta nya besökare till de sidor och den information som de söker. Skogsreviret ville ha en enkel webbsida som klart och tydligt kan leda nya och tidigare besökare vidare.

Till börja med ville de ha en huvudbild som fungerar som en *call to action* sektion. I denna sektion finns det en huvudbild, samt två knappar som fungerar som länkar. Den första knappen skulle leda till en sida som berättar allmänt om Skogsreviret och den andra knappen skulle gå direkt till kontaktsidan där man kan hitta kontaktuppgifter.

Efter huvudbilden ville de ha en sektion som visar de tjänster som Skogsreviret erbjuder. Varje tjänst skulle ha en egen bild samt en länk direkt under bilden för att en besökare skall kunna ta sig vidare till den valda tjänstens sida.

Man ville även att nyheter skulle visas på framsidan. I detta fall var de enda kraven att en nyhet skulle ha en bild, rubrik, kort text samt en "läs mera" knapp som leder till nyhetens sida där hela texten kan läsas. Jag fick bestämma designen av dessa nyhetslådor själv och kommer att berätta mera om detta i följande kapitel.

Sista kravet på startsidan var en kontaktsektion som skulle inkludera ett kontaktformulär, kontaktuppgifter och en karta var huvudkontoret finns. Kontaktformuläret skulle ha fält som samlar in för- och efternamn, epost (inte obligatoriskt för att skicka in formuläret), telefonnummer, rubrik och ett meddelande. Kontaktuppgifterna skulle visas i en låda där huvudkontorets adress, telefonnummer, epost och en länk till kontaktsidan.

3.3 Tvåspråkighet

Som jag nämnde tidigare så var ett av kraven att webbsidan skulle visas på båda inhemska språken. Varje sida skulle ha en egen version på båda språken. I huvudmenyn ville man även att språkval skulle placeras i form av flaggor. Orsaken till att man föredrog flaggor framför text var att flaggorna ansågs vara tydligare och fastnar lättare i

en besökares ögon. Ett av kraven med språkvalet var att om man befinner sig på en sida och byter språk så kommer man direkt till den andra språkversionen av sidan och inte till startsidan. Detta problem hade man bland annat på den gamla webbsidan. Viktigt var även att välja ett insticksprogram som tillåter översättning direkt i användargränssnittet så att personalen kan skapa ett inlägg att visas på båda språken.

3.4 Elektroniska formulär för insamling av information

På den gamla webbsidan skedde en stor del av beställningarna genom att manuellt ladda ner en pdf-fil, fylla i och skicka tillbaka till Skogsreviret. På den nya sidan var kraven att alla beställningar och meddelanden ska skötas elektroniskt direkt från webbsidan för att underlätta genomgång av ansökningar och för att hålla informationen organiserad och på samma plats. I de olika formulären ville man också att vissa fält skulle vara obligatoriska för att undvika att halvfärdiga svar skickas in. För att undvika spam ville man även att ett anti-spam system skulle installeras på formulären.

Eftersom många skogsägare inte har en egen epost ville man att alla epost fält skulle ställas in som valfria. I och med detta ville man inte heller ha något automatiskt epost svar utan sköta kontakt av kunderna per telefon, eller per epost ifall fältet fyllts i.

3.5 Responsivitet

Ett av de viktigaste kraven på den nya webbsidan var att sidan skulle vara responsiv och optimerad för olika skärmstorlekar som en modern webbsida borde vara. En stor del av Skogsrevirets medlemmar använder sig av läsplattor och smarttelefoner så därför ville man att webbsidan skulle vara lättnavigerad oberoende av skärmstorlek. Skogsreviret ansåg inte att det var nödvändigt att optimera webbsidan för större resolutioner så som 4K (3840x2160 pixlar). Den största skärmstorleken som testats är 1080p (1920x1080 pixlar).

3.6 Självadministrering av webbsidan

För att inte behöva kontakta mig då man skulle uppdatera webbsidans innehåll ansåg man att webbsidans användargränssnitt skulle utvecklas så att personalen på Skogsreviret själva kunde uppdatera webbsidans innehåll. Exempel på innehåll som skulle skötas av dem själva är nyheter, textbaserad info som skrivs ut på olika undersidor och administrering av kontaktformulär ifall man ville lägga till ett fält eller ta bort ett fält. Jag fick även i uppgift att lära in funktionaliteten då man lägger till innehåll åt ledningen.

4 VAL AV BYGGSTENAR

I detta kapitel behandlar jag valen av byggstenar och motiverar varför jag valt just dem.

4.1 Val av WordPress tema

När det var tid för att välja tema så sökte jag mig till <https://themeforest.net/> för att hitta ett premium-tema som skulle passa in. Redan då visste jag att jag inte ville ha ett tema som var så färdigt som möjligt, utan ett enkelt tema med basfunktionalitet för att sedan bygga den önskade designen på med hjälp av CSS ramverk och insticksprogram.

Det temat som jag bestämde mig för att välja var ett tema som heter X-theme. Det är ett tema som är utvecklat av Themeco. Orsaken till valet var nog mest för de betalda insticksprogram som följde med temat som i sin tur underlättade utvecklingen av webbsidan. Temat hade även en god och enkel grund att bygga på med responsiva egenskaper. Ett alternativ som jag kollade på var ett tema vid namn The7, men eftersom det inte innehöll alla de insticksprogram som jag sökte efter så valde jag att arbeta med X-theme. Dessutom var jag redan från början mera erfaren med X-theme.

4.2 Val av insticksprogram

4.2.1 Advanced Custom Fields Pro

Advanced Custom Fields som ingår i X-themes sortiment är ett premium-insticksprogram som tillåter en skapa extra innehållsfält på användargränssnittet för sidor och inlägg. Man kan sedan skriva ut dem på en valfri sida genom att lägga till en funktion i en bestämd layout fil. Denna process tas upp mera i kapitel fem. Advanced Custom Fields Pro är en förbättring av Advanced Custom Fields som är ett gratis insticksprogram. Det som bland annat skiljer dem åt är att Advanced Custom Fields Pro innehåller flera olika sorters fält med avancerade egenskaper. Jag hade tidigare hört mycket gott om Advanced Custom Fields Pro men aldrig använt det förrut. Jag valde det eftersom det visade sig vara lättanvänt och effektivt för att skriva ut information. Med hjälp av detta är det dessutom lättare för Skogsrevirets ledning att byta ut och skapa nytt material. (Advanced Custom Fields Pro 2018)

Ett alternativ till att skriva ut information som jag även funderade på heter Cornerstone. Skillnaden mellan Cornerstone och Advanced Custom Fields Pro är att Cornerstone endast sköts från X-themes användargränssnitt och låter en lägga till information på en valfri sida med hjälp av *drag&drop* funktion. Man drar det valda elementet in i ett rutnäts system.

Orsaken till varför jag valde att inte använda Cornerstone var för att det inte tillåter lika mycket anpassning som Advanced Custom Fields Pro. Cornerstone har en massa färdiga block som kan editeras men det finns begränsningar. Med Advanced Custom Fields Pro kan man göra vad man vill med vilket utseende som helst bara man har kunskapen att skapa en layout fil och läsa in information i den. I och med att personalen inte har något behov att röra webbsidans kod så tyckte jag själv att det skulle vara mycket tydligare att använda då man fyller i information i användargränssnittet. Risken att en sida förstörs är även mycket mindre då man med Advanced Custom Fields Pro endast har tillgång till meta fält i användargränssnittet. Om personalen på Skogsreviret skulle använda Cornerstone finns det risk för att något förstörs på sidan ifall t.ex. något block raderas i misstag.

4.2.2 Polylang

Som nämnt tidigare var ett av kraven på den nya webbsidan att den skulle skapas på både svenska och finska. Polylang är ett insticksprogram som tillåter översättning av sidor och strängar på en WordPress hemsida (Nsp-code & Tdgu 2018). Det ingår inte i X-themes sortiment.

Jag hade endast ett annat alternativ för att översätta webbsidan och det var ett insticksprogram vid namn WordPress Multilingual Plugin. Jag läste mig in på det på insticksprogrammets dokumentation (WPML 2018) men kom fram till att det skulle vara lättare för Skogsreviret att skapa flerspråkiga sidor med hjälp av Polylang då det endast sker med ett knapptryck för att få upp en ny sida att editera på annat språk.

Underlättande i inläringen var även att jag använt mig av Polylang fler gånger och aldrig WordPress language switcher och därför valde jag att använda Polylang.

4.2.3 Ninja Forms

Ninja Forms är ett insticksprogram som låter en skapa dynamiska formulär som integreras in på valfri plats på en sida med hjälp av en såkallad *shortcode*. Det ingår inte i X-themes sortiment.

Då jag började att arbeta på specifikationen runt kontaktformulären så sökte jag mig till ett insticksprogram vid namn Contact Form 7. Planen med kontaktformuläret var att man skulle ställa in epost fältet som frivilligt så att de som fyllde i epost fick en bekräftelse till den valda adressen och sedan blev kontaktade av Skogsreviret. De som däremot inte hade en epost behövde inte fylla i det fältet och blev direkt kontaktade av Skogsreviret per telefon. Contact form 7 stödde inte detta utan för att implementera en bekräftelse så krävdes att epost fältet var obligatoriskt. Ninja Forms som jag sedan valde istället behövde inte eposten som obligatorisk utan kände av om epost fältet var ifyllt eller tomt och skickade sedan en bekräftelse baserat på detta. Om Skogsreviret skulle vilja ändra på något av formulären är dessutom Ninja Forms lättare att editera med då det använder sig av ett *drag & drop* system för att dra olika fält på plats i formuläret. Contact Form 7 byggs istället upp med shortcodes internt.

4.2.4 Slider Revolution

Slider Revolution är ett premium insticksprogram gjort av Themepunch som tillåter en att skapa dynamiska sliders, karuseller och t.o.m. hela sidor med hjälp av *drag & drop* funktion. Det ingår även olika animationseffekter som man kan lägga till de olika elementen som man använder. Exempel på dessa är intonings effekter, glid effekter från olika håll och även *parallax* effekter är möjligt.

Jag valde att använda detta för att det ingår i X-themes sortiment av insticksprogram och visste redan från början att webbsidan skulle ha en huvudslider. Slider Revolution tillämpades även på några andra undersidor.

4.2.5 Post Types Order

Post Types Order är ett insticksprogram som tillåter ändring av ordningen av valfri inläggstyp i användargränssnittet. Alla inläggstyper visas sedan i den ordning som man ordnat dem i, förutsatt att man inte har lagt till en parameter som bestämmer ordningen av inläggstyper skilt i WordPress loop. Jag valde denna plugin främst för att ledningen på Skogsreviret själva skulle kunna ändra på ordningen av diverse inläggstyper. Jag valde programmet för att jag hade använt mig av det förrut och visste att det var tillräckligt enkelt för nya användare att dra nytta av. Jag valde det även för att hålla ordning på t.ex. innehållet på personalsidorna så att de högre uppsatta dyker upp först utan att behöva spendera för mycket tid på det. Det ingår inte i X-themes sortiment av insticksprogram.

4.3 Bootstrap

För att snabbare göra webbsidan responsiv bestämde jag mig för att använda Bootstrap som är ett gratis CSS och Javascript ramverk. Med hjälp av Bootstrap kan man göra sin webbsida baserad på ett rutnätssystem som ändras baserat på vilka inställningar som man gett till ett HTML element. Varje rad har 12 stycken kolumner i bredd så man kan ge bredden på ett element allt från ett till tolv. Jag tar upp implementering av Bootstrap och hur det fungerar i praktiken under startsidans implementering.

Jag valde Bootstrap för att jag använt mig av det förrut och var medveten om hur effektivt det fungerar. Jag gjorde ingen studie över detta utan valde det av egen erfarenhet.

5 RESULTAT OCH IMPLEMENTERING AV KRAV

I detta kapitel kommer jag att berätta resultat på hur jag löst kravspecifikationerna och vilka av de valda byggstenarna som har använts i processen.

5.1 Navigering

Webbsidans navigering har implementerats nästan bara med hjälp av WordPress standard användargränssnitt för skapande av menyer. Man kan skapa menyer och lägga till länkar genom att gå till utseende länken i användargränssnittet och klicka på menyer. Där kan man sedan skapa en ny meny och lägga till länkar med hjälp av vertyget på vänster sida.

Jag började med att skapa huvudmenyn dit jag efterhand lade till de sidor som jag skapat. Med hjälp av Polylang kunde jag även lägga till två stycken flaggor (svensk och finsk) för att låta besökare välja språk från huvudmenyn. Jag lade till dem genom att kryssa för ”språkväljare” i verktyget till vänster och klicka på lägg till i meny. Jag placerade flaggorna längst ner vilket betyder att de dyker upp längst till höger på meny balken. För att välja det rätta språket att visa rätt meny på scrollade jag längst ner på menyens editeringssida och kryssade för att menyn endast skall visas på svenska. Sedan skapade jag en till meny där jag lade till finska meny länkar och språkval. Tillsist kryssade jag för att den skulle visas endast på finska sidan. För att skapa de planerade sidomenyerna använde jag mig av samma metod som för huvudmenyn.

Ovanför menyn kan man med hjälp av X-themes användargränssnitt välja att visa en *topbar* dit man kan placera eget innehåll. Jag valde att använda den för kontaktuppgifter och sökrutan som efterfrågats. Problem som jag stötte på var att sökrutan inte visade allt innehåll som den borde visa vid en viss sökning. Av någon anledning valde WordPress att inte visa egen skapade inläggstyper. Även fast jag inkluderade alla inläggstyper i

sökkriterierna så visades inte de egenskapade inläggstyperna. Jag lyckades hitta en lösning på Stack Exchange som tipsade om att göra flera skilda sökkriterier och sedan kombinera dem i samma sökning med hjälp av en funktion som heter `array_merge` (Stack Exchange 2018). Exempel på koden finns nedan. Jag skapade även en sökruta på kontaktsidan som man kan söka på personalens medlemmar. Där uppstod inte samma problem då jag endast sökte på personalens inläggstyp.

```
$q1 = new WP_Query( array(
    'post_type' => array('page', 'post'),
    'posts_per_page' => -1,
    's' => $search_term
));

$q2 = new WP_Query( array(
    'post_type' => array('contacts'),
    'posts_per_page' => -1,
    's' => $search_term,
    'order' => 'ASC',
));

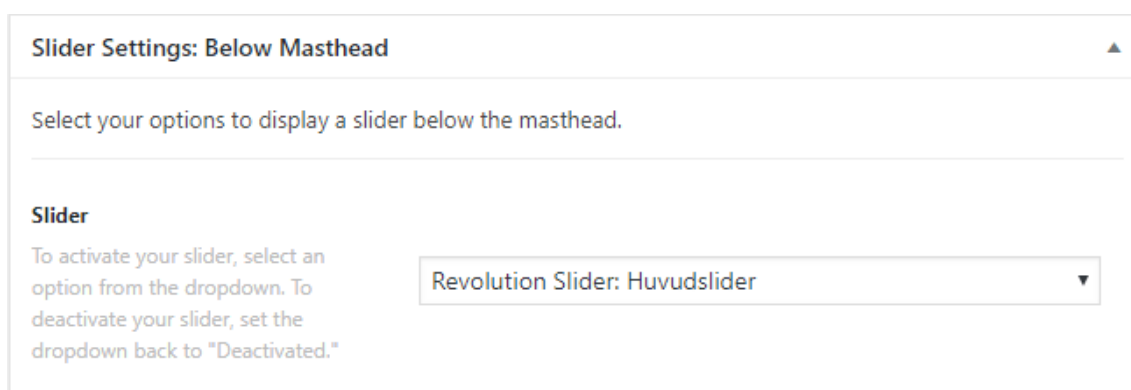
$q3 = new WP_Query( array(
    'post_type' => array('styrelsen'),
    'posts_per_page' => -1,
    's' => $search_term,
    'order' => 'ASC',
));

$query = new WP_Query();
$query->posts = array_unique( array_merge( $q1->posts, $q2->posts, $q3->posts ), SORT_REGULAR );
$query->post_count = count( $query->posts );
```

5.2 Startsidan

Startsidan är uppbyggd av flera olika insticksprogram. Den första sektionen, det vill säga huvudslidern är uppbyggd i Slider Revolution. Jag skapade slidern genom att gå till Slider Revolution i användargränssnittet och klicka på new slider. Sedan klickade jag på slider editor och fyllde i de inställningar som krävdes. Efter det klickade jag på slide editor. Där lade jag till en bakgrundsbild, två stycken text lådor och två stycken knappar.

För att skriva ut huvudslidern gick jag till editeringsvy för startsidan. X-theme har ett inbyggt fält på varje editeringssida som tillåter en valfri slider att visas antingen ovanför eller under huvudmenyn. Där valde jag den nyskapade slidern och sparade sidan (Figur 4). För att även få en finsk version av slidern skapade jag en kopia av slidern som jag lade till på den finska startsidans editeringsvy.



Figur 4. Fältet som man kan lägga till en slider med.

Under processen stötte jag på problem med responsiviteten. Huvudslidern såg bra ut i skrivbordsmiljö men i mobilen var texten alltför stor och klumpade ihop sig. Jag löste problemet enbart inom Slider Revolution. Programmet har ett inbyggt system som förhandsvisar slidern i fyra olika skärmstorlekar från skrivbordsmiljö till mobil. Inom varje miljö kan man positionera ett valfritt elementet och välja i vilken miljö som det skall visas. Detta gäller dock endast positionen, vilket betyder att om man ändrar storleken på texten så ändrar den i alla skärmstorlekar. Därför skapade jag nya element för de textstycken som inte såg bra ut i mobilen. Jag ändrade textstorleken på det nya

elementet. Sedan behövde jag bara välja i vilka miljöer de två olika texterna skulle visas. Ett exempel på hur startsidan ser ut då man besöker den syns nedan (Figur 5).



Figur 5. Vyn då man besöker startsidan.

Den andra sektionen som visar Föreningens åtta huvudtjänster skapades med hjälp av Advanced Custom Fields och Bootstrap. Bootstrap kunde installeras genom att placera dess filer inom temats filträd och sedan inkludera dem med att placera kod i functions.php filen. Jag inkluderade Bootstraps filer med hjälp av WordPress egna instruktioner för att inkludera CSS och Javascript filer (WordPress.org).

Eftersom föreningen har åtta stycken huvudtjänster skulle de placeras som fyra i bredd på två rader och sedan flyta under varandra då skärmstorleken krymper. För att implementera detta omslöt jag varje enskild tjänst i ett <div> element som jag sedan gav Bootstrap klasser åt. Exempel på div elementet från den första tjänsten visas nedan.

```
<div id="service-1" class="col-xl-3 col-lg-4 col-md-6">
```

De olika siffrorna i klasserna innebär bredden på elementet av de 12 stycken kolumner som finns. Klassen för mobilskrmar är inte tillagd då Bootstrap känner igen när det är dags att placera alla element under varandra.

Sedan gick jag till Advanced Custom Fields meny i användargränssnittet och skapade en ny fältgrupp. I fältgruppen lade jag till åtta stycken bilder och länkar och valde att den skulle visas på startsidans användargränssnitt. Där fyllde jag i de bilder som jag fått och de bestämda rubrikerna, samt länkade dem till sina undersidor. För att det innehåll som skapats skall kunna visas på en sida måste man läsa in dem med hjälp av en funktion som placeras på valfri plats i sidans template fil. Det är viktigt att kontrollera namnet på fältet och jämföra det med namnet i funktionen. Om det finns ett stavfel läses inte fältets information in. Exempel på koden:

```
$service_rubrik1 = get_field('service_1_rubrik');
```

Med hjälp av den funktionen sparas informationen från fältet med det valda namnet i en variabel som sedan kan läsas in på valfri plats. I exemplet är variabeln en länk som läser in texten som skall visas. Alla variabler placerades i sina respektive Bootstrap element. Exempel på enskilt element av en tjänst som placerats inom ett Bootstrap element:

```
<div id="service-1" class="col-xl-3 col-lg-4 col-md-6">
  <div class="service-icon">
    <div class="service-icon-wrap">
      
    </div>
  </div>
  <p class="service-heading"><a href="<?php echo $service_rubrik1['url']; ?>" target="<?php echo $service_rubrik1['target']; ?>"><?php echo $service_rubrik1['title']; ?></a></p>
</div>
```

Mera information om Advanced Custom Fields kan läsas från dokumentationen från (Advanced Custom Fields 2018).

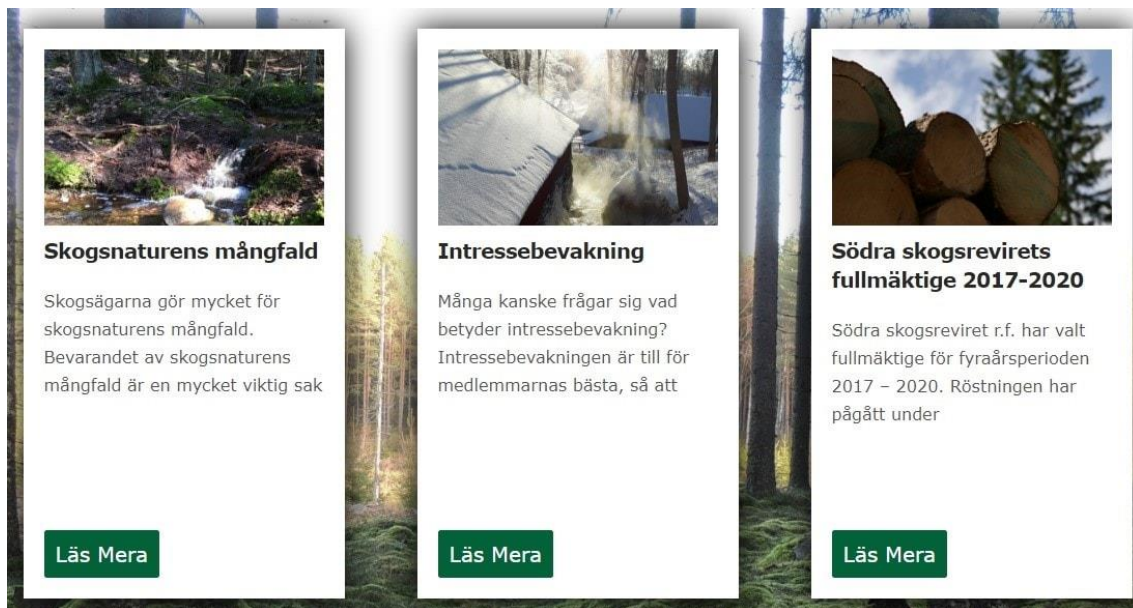
I den sektionen stötte jag inte på problem. Implementeringen gick snabbt och smidigt. Exempel på sektionen finns nedan(Figur 6).



Figur 6. Startsidans sektion som visar huvudtjänster.

Den tredje sektionen som består av tre nyhetsartiklar använder sig inte av något insticksprogram. Endast Bootstrap har använts för att få ett responsivt utseende. Jag implementerade responsiviteten på samma vis som i föregående sektion men med inställningen att endast tre stycken artiklar visas på samma rad. För att skriva ut varje nyhet har jag använt mig av en WordPress loop som skriver ut de tre senaste nyheterna. Exempel på sektionen finns nedan (Figur 7). Det problem som jag stötte på var att ”läs mera” knappen inte var i bottner av en nyhetslåda från början utan rakt under texten. Dessutom var texten på knappen för liten. Jag justerade detta med hjälp av CSS. Koden kan ses nedan.

```
.read-more-link {  
    font-size: 20px;  
    position: absolute;  
    bottom: 20px;  
}
```



Figur 7. Startsidans sektion som visar nyheter.

I den sista sektionen finns ett kontaktformulär, kontaktinformation och en karta på föreningens huvudkontor. De är uppdelade i två kolumner så att kontaktformuläret ligger till vänster och de andra två elementen till höger.

Jag skapade formuläret i ett insticksprogram som heter Ninja Forms. I formulärets användargränssnitt lade jag till de fält som efterfrågats och implementerade även reCaptcha som är ett anti-spam system. För att sedan skriva ut formuläret på webbsidan använde jag mig av en *shortcode* som placerades i ett fält skapat av Advanced Custom Fields. Jag läste sedan in fältet till vänster för att visa formuläret på sidan. Både kontaktinformation och kartan skrivs ut med hjälp av Advanced Custom Fields. Exempel på hur kontaktsektionen ser ut finns nedan (Figur 8).

Förnamn och efternamn *

Epost

Telefon *

Rubrik *

Meddelande *

☐ En ole robotti

reCAPTCHA
Täsmuoto - Ohjeet

**SKOGSVÅRDSFÖRENINGEN
SÖDRA SKOGSREVIRET**

Mannerheimgatan 9-11 A, 06100 BORGÅ

Telefon: 019 580 993

Epost: lovisa@revir.org

Våra kontor >>

Kartta Satelliitti

Vanha Porvoo

Vanha Brunbergin makeistehdas

Runebergin talo

Porvoon VPK:n talo

Pelaamo Porvoo Lund

Itä-Uudenmaan Rakennussuojelusäätiö

Johan Ludvig Runebergin patsas

Porvoo

Kaupunginpuisto

Google

Karttatiedot ©2018 Google Käyttöehdot Ilmoita karttavirheestä

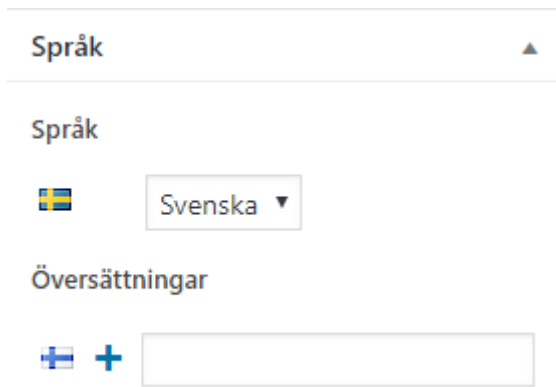
Figur 8. Startsidans kontaktsektion.

5.3 Tvåspråkighet

Då jag fick kravet att webbsidan skulle visas på två olika språk visste jag att det bästa sättet att gå tillväga var att söka upp ett insticksprogram för att sköta översättningar.

I detta fall använde jag mig av Polylang som är ett populärt insticksprogram för översättning av webbsidor.

Översättningen av webbsidan gick snabbt och smidigt och det mesta kunde översättas genom standard översättningsfunktionen i Polylang, det vill säga genom att skapa en kopia av den svenska sidan och sedan editera det valda språkets version. Detta gör man genom att på en sidas användargränssnitt skriva in ett namn i fältet och klicka på plus tecknet (Figur 9).



Figur 9. Polylangs användargränssnitt för språkval på editeringssidan.

Då tas man till en ny sida med tomma fält som fungerar som sidan för det valda språket. I mitt fall fanns det endast ett alternativ, det vill säga finska.

Översättning av menyer använder samma princip som det ovannämnda tillvägagångssättet. Jag skapade kopior på alla menyer för att sedan välja vilket språk de skulle visas på. Endast några få ord och meningar måste översättas med hjälp av strängöversättningsmetoden som betyder att man omger det valda ordet att översätta med en funktion och sedan registrerar det i functions.php filen.

Exempel på funktion som indikerar att strängen skall översättas:

```
<?php pll_e('Revirets tjänster'); ?>
```

Exempel på registreringen av strängen i functions.php:

```
pll_register_string('Revirets tjänster', 'Revirets tjänster');
```

För att översätta det registrerade ordet går man till Polylangs sektion i användargränssnittet där man kan fylla i det översatta ordet. Jag stötte endast på problem när vissa specifika strängar skulle översättas så som "läs mera" knappen under inlägg som är inbyggd i temat. Kodediterings programmet som jag använder som heter Atom har en inbyggd funktion som låter en söka genom hela projektmappen så jag sökte

på order ”Read more”. Då jag hittat funktionen som skapar knappen registrerade jag strängen i Polylang och översatte den med strängöversättning.

5.4 Elektroniska formulär för insamling av information

För att uppfylla kravet på kontaktformulären var det bästa tillvägagångssättet ett av de populära insticksprogrammen som skapar dynamiska formulär. I detta fall använde jag mig av Ninja Forms. Jag började med att skapa ett nytt formulär för att använda det på startsidan och testa det. Då jag lagt till de efterfrågade fälten skrev jag ut formuläret med hjälp av en *shortcode* som placerades i ett Advanced Custom Fields fält. När formuläret var testat och jag visste att det fungerade skapade jag resten av formulären och lade till dem på samma vis på sina egna sidor.

För att formuläret skulle vara skyddat från automatisk ifyllning som get möjlighet för spam bestämde jag mig även för att lägga till reCaptcha på varje formulär. För att lägga till en reCaptcha måste man först gå till googles reCaptcha sida och ansöka om en platsnyckel och en hemlig nyckel. Nycklarna kan hämtas från googles egen sida för reCaptcha: <https://www.google.com/recaptcha/intro/>. I inställningarna för Ninja Forms finns färdiga fält för att fylla i nycklarna. Då nycklarna sparats navigerade jag till varje formulär och lade till ett reCaptcha block på samma vis som man lägger till ett textfält i Ninja Forms.

Ett problem som uppstod med Ninja Forms utseende var att alla textfält inte var i lagom höjd så genom att lägga till en CSS klass för varje fält och justera höjden på fältet med CSS kod så blev designen snyggare och tydligare. Exempel på koden som justerar höjden av *input* elementen i Ninja Forms:

```
.nf-field-element input {  
    height:30px!important;  
}
```

Knappen som skickar in formuläret hade inte en passande färg så jag justerade även färgen på den med hjälp av samma metod (Se koden nedan).

```
.form-button {  
    background-color: rgba(130,190,30,0.8);  
    color: #fff;  
}
```

5.5 Responsivitet

I den bas som medföljer temat var redan huvudmenyn och footern responsiva så jag har inte behövt fokusera så mycket på dem. Angående menyerna stötte jag endast på ett responsivets problem som gjorde att antalet länkar tvingade menybalken att bli på två rader. Jag fixade detta genom att minska på avståndet mellan menylänkarna samt minska avståndet mellan bokstäverna i varje länk. För säkerhets skull minskade jag även på bokstävernas storlek i varje länk. Det gjorde jag genom att lägga till denna CSS kod under menylänkarnas klass i temats style.css fil.

```
font-size: 18px;  
padding-left: 15px;  
padding-right: 15px;  
letter-spacing: 0px;
```

För att det innehåll som finns mellan ovannämnda sektioner skulle fungera responsivt har jag för det mesta använt mig av Bootstrap. Med hjälp av detta front-end CSS ramverk har jag skapat flera olika layout filer där jag har gett Bootstraps klasser åt elementen.

Det största problemet som jag stötte på var att efter att jag delat ut Bootstraps klasser åt olika element så visades en stor del av webbsidans element som alltför smala i en läsplatta. Exempel på problemet nedan i en iPad (Figur 10).

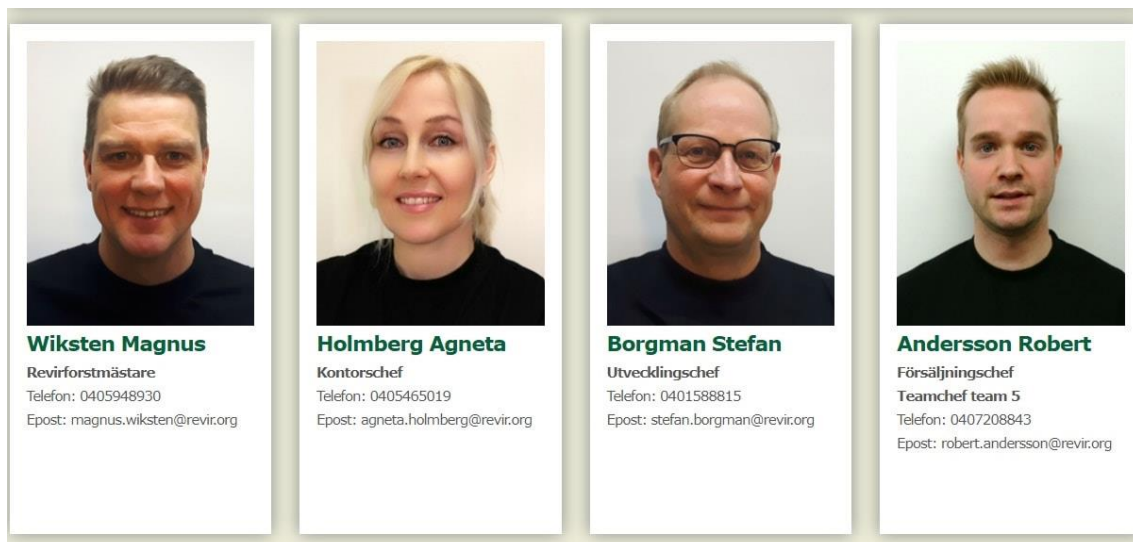


Figur 10. Utseendet på en undersida före(vänster) och efter (höger) responsiviteten fixats.

Jag märkte då att jag definierat övergången från t.ex. tre kolumner till full bredd alltför sent. Jag fixade problemet genom att ändra på CSS klassen för de element som påverkats. Då jag ändrat på klassen visades de element som varit problematiska i full bredd.

Jag stötte även på problem då jag skapade personalsidorna. Eftersom de bilder jag fått till förfogande hade olika bildförhållande blev personallådorna i olika höjd vilket inte såg bra ut. Detta problem uppstod även då man skulle ladda upp en nyhet då bilderna som skulle laddas upp också oftast hade olika bildförhållande.

Jag löste problemet genom att använda mig av ett tillägg för Advanced Custom Fields som, heter Advanced Custom Fields: Image Crop Add-on, skapat av Anders Thorborg. Med hjälp av detta tillägg kan man skära bilden till ett bestämt bildförhållande i användargränssnittet då man laddar upp bilden. På så vis fick alla bilder samma förhållande mellan bildens bredd och höjd vilket såg mycket snyggare ut (Figur 11).



Figur 11. Innehåll på personalsida efter bildförhållandet justerats.

5.6 Självadministrering av webbsidan

För att göra självadministrering så lätt som möjligt har jag använt mig av Advanced Custom Fields Pro för att skapa meta fält som skriver ut informationen på webbsidan. Jag har även använt mig av Post Types Order så att man skall kunna ändra ordningen på olika inlägg inom en specifik inläggstypsgrupp.

Jag stötte inte på problem när självadministreringen skulle planeras men det tog ändå en stund att komma på namn till olika meta-fält för att göra webbsidan så enkel som möjligt att administrera. Jag ändrade även användargränssnittets språk för att de lättare skall kunna förstå och navigera. Detta gjorde jag genom att i användargränssnittet gå till Allmänt under inställningar. På den sidan finns ett alternativ för att ändra språk på webbsidan.

Jag skrev även ner en guide som berättar hur man lägger till och tar bort olika inläggstyper samt hur man använder sig av Polylangs översättning då nya inlägg skulle skapas. Guiden presenteras inte i detta arbete och fungerar endast som en manual för ledningen på Skogsreviret.

5.7 Sammanfattning av resultat

Då alla krav var implementerade fick Skogsrevirets ledning testa sidan och bedöma hur bra den motsvarar kravspecifikationerna. Efter att Skogsreviret hade kontrollerat webbsidan ansågs den som klar att publiceras efter några små utseende korrigeringar med hjälp av CSS. De byggstenar som har använts under implementeringsprocessen har uppfyllt kravspecifikationerna utan större problem och webbsidan kan nu kallas en modern webbsida.

X-theme har lämpat sig utmärkt för utvecklingen av den nya webbsidan. Eftersom det har en kraftig bas utan en massa färdigt innehåll var det lätt att lägga till eget innehåll. De insticksprogram som använts har uppfyllt sin uppgift och valet på de moduler som använts är något som kunde rekommenderas till en läsare som planerar att skapa en ny webbsida i WordPress.

Speciellt användbart och överraskande var användningen av Advanced Custom Fields. Med sitt enkla användargränssnitt och sätt att skriva ut information har nästan allt innehåll skrivits ut med hjälp av det.

Polylang fyllde sin uppgift så att allt innehåll kunde översättas snabbt. Det som även är en fördel med Polylang är att det fungerar så att även en nybörjare kunde börja använda det efter en kort guide vid översättning av tex. en nyhet.

Slider Revolution har fungerat utmärkt i jämförelse med kravspecifikationen på startsidans huvudslider. Även fast man skulle kunna skapa något liknande med hjälp av t.ex. Advanced Custom Fields var det värt att testa.

Eftersom elektroniska formulär har implementerats med hjälp av Ninja Forms kan nu skogsägare skicka in en ansökan eller en beställning direkt på webbsidan istället för att ladda ner en pdf-fil som man fyller i och skickar tillbaka. Detta gör det lättare för både kunder och personal genom att minska på pappersarbete då allt samlas i elektronisk form. Informationen kommer även snabbare in till kontoret eftersom den inte måste postas mera.

Post Types Order är det minst viktiga för att webbsidan skall fungera men eftersom det underlättar Skogsrevirets personal med att organisera innehåll användes det.

6 DISKUSSION OCH SLUTSATS

Som tidigare nämnt var målet att skapa en så användarvänlig hemsida som möjligt med stöd för olika skärmstorlekar samt att en del av personalen på Skogsreviret skulle ha möjlighet att uppdatera innehållet på webbsidan.

I jämförelse med den gamla webbsidan är nu innehållet ordentligt strukturerat och navigeringen är mycket enklare för en person som besöker webbsidan första gången.

Ledningen på Skogsreviret är nöjda med min insats och webbsidan motsvarar de krav som gavs under vårt första möte. Skogsrevirets ledning har även visat webbsidan åt sina anställda och folk är nöjda med den.

I jämförelse med min kunskap innan webbsidan skapades så är jag nu mycket mera kunnig och flytande inom WordPress och innehållshanterings system i sin helhet. Jag har även kommit i kontakt med WordPressmoduler som jag aldrig använt mig av förrut som visade sig vara effektiva. För att lära mig mera om olika innehållshanteringssystem borde jag skapa en webbsida i t.ex. Joomla! eller Drupal för att inte fastna på att använda endast WordPress.

Valet av byggstenarna är något som är aktuellt just nu, men det är mycket möjligt att en del av dem har föråldrats om t.ex. 5 år. Det är därför mycket möjligt att jag kommer att få byta ut någon byggsten i framtiden ifall en av dem blir t.ex. klassad som osäker.

Överlag är jag nöjd med min insats i skapandet av webbsidan och skulle rekommendera WordPress till andra som är intresserade att bygga en moden webbsida som uppfyller kravspecifikationer.

KÄLLOR

Advanced Custom Fields Pro 2018, Tillgängligt:

<https://www.advancedcustomfields.com/pro/> Hämtad 01.04.2018

Advanced Custom Fields 2018, Documentation. Tillgänglig:

<https://www.advancedcustomfields.com/resources> Hämtad 27.04.2018

Astra 2018, Security Comparison of CMS: WordPress vs Drupal vs Joomla. Tillgänglig:

<https://www.getastra.com/blog/cms/security-comparison-of-cms-wordpress-vs-drupal-vs-joomla/> Hämtad 20.04.2018

Codex 2017, Child Theme. Tillgänglig:

https://codex.wordpress.org/Child_Themes Hämtad 15.03.2018

Chouby. 2018, Polylang, Frédéric Demarle. Tillgänglig:

<https://wordpress.org/plugins/polylang/> Hämtad 28.03.2018

IT-ord. 2018a, Innehållshanteringssystem. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/ord/innehallshanteringssystem/> Hämtad 04.04.2018

IT-ord. 2018b, CSS. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/ord/css/> Hämtad 04.04.2018

IT-ord. 2018c, Insticksprogram. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/ord/insticksprogram/> Hämtad 04.04.2018

IT-ord. 2018d, Kravspecifikation. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/ord/kravspecifikation/> Hämtad 04.04.2018

IT-ord. 2018e, HTML. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/?s=HTML> Hämtad 04.04.2018

IT-ord. 2018f, PHP. it-ord.idg.se Tillgänglig:

<https://it-ord.idg.se/ord/php/> Hämtad 04.04.2018

Ninja Forms 2018. wordpress.org. Tillgänglig:

<https://wordpress.org/plugins/ninja-forms/> Hämtad 01.04.2018

Nsp-code & Tdgu. 2018, Post Types Order, NSP Code. Tillgänglig:

<https://wordpress.org/plugins/post-types-order/> Hämtad 28.03.2018

Themepunch. 2018, Slider Revolution Tillgänglig:

<https://revolution.themepunch.com/> Hämtad 01.04.2018

Stack Exchange 2018, wordpress.stackexchange.com. Tillgänglig:

<https://wordpress.stackexchange.com/questions/55519/can-i-merge-2-new-wp-queryvariable-s> Hämtad 15.01.2018

WordPress Codex 2018, About WordPress. Tillgänglig:

<https://wordpress.org/about/> Hämtad 01.04.2018

WordPress.org 2018, Including CSS & Javascript. Tillgänglig:

<https://developer.wordpress.org/themes/basics/including-css-javascript/>

Hämtad 27.12.2018

WordPress support 2018, Shortcodes. Tillgänglig:

<https://en.support.wordpress.com/shortcodes/> Hämtad 25.04.2018

WPML 2018, Documentation. Tillgänglig:

<https://wpml.org/documentation/> Hämtad 20.04.2018

